Charles H. Roth, Jr. (S'54–M'56) was born in St. Paul, MN, on December 5, 1932. He received the B.E.E. degree from the University of Minnesota, Minneapolis, in 1955, the S.M. and E.E. degrees from the Massachusetts Institute of Technology, Cambridge, in 1957 and 1959, respectively, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1962.

In 1961 he joined the faculty of the University of Texas at Austin, where he is currently serving as Professor of Electrical Engineering. His current teaching and research interests include microcomputer systems and the theory and design of digital systems. He is the author of several textbooks, the most recent of which is *Fundamentals of Logic Design.*

Dr. Roth is a member of the American Society for Engineering Education, the Association for Computing Machinery, Tau Beta Pi, Eta Kappa Nu, and Sigma Xi.

John S. Fernando (S'82) was born in Colombo, Sri Lanka, on October 11, 1955. He received the B.Sc. and M.S. degrees from the University of Sri Lanka and the University of Texas, Austin, in 1978 and 1983, respectively.

He worked as an Engineer at the National Engineering Research and Development Center of Sri Lanka in 1978, and later joined the staff of the University of Moratuwa, Sri Lanka. He is currently with Burroughs Corporation, Mission Viejo, CA.

A Microcomputer-Based Reading Aid for Blind Students

OLUWOLE REMI OMOTAYO, MEMBER, IEEE

***Abstract*—In schools and colleges for the blind, Braille is the popular medium of accessing textual educational information. However, in situations when the text does not contain diagrams, graphs, or maps, converting the text to speech would increase the reading speed of the blind student considerably. In this paper, the techniques used in designing and evaluating the performance of a microcomputer-based system which automatically reads out textual information in synthetic speech is described.**

Introduction

MANY phonetically programmable synthesizers are now commercially available, and rapid development in microelectronics has made it possible for speech synthesizers to be realized in small sizes. At the extreme is the realization of a whole speech synthesizer circuitry in a single integrated circuit chip. Examples of such single-chip speech synthesizers are Votrax SCO1 [1] and TMS5220 [2]. Both are capable of handling unlimited vocabulary.

Programming a synthesizer to speak a given text, however, is a complex process, which in the early days of text-to-speech systems could only be efficiently carried out on large computers. First, applying some letter-to-sound rules, the text is converted to the basic units of speech (phonemes) [3]. These phonetic units are then fed into a speech synthesizer which constructs the sound waves required to generate the speech. In addition, the text is simultaneously analyzed syntactically, by applying some rules of grammar, in order to determine the positions of appropriate pauses or stresses.

However, microprocessors have now made it possible to realize a whole computer in a couple of integrated circuit chips as a microcomputer [4], [5]. Thus, the processing capability and speed required to convert text into speech in real time can now be conveniently provided by a microcomputer. This paper describes a method of designing such a portable system as a reading aid for the blind, especially for those who are students.

In some speech synthesizers, compressed digitized human speech is stored for eventual retrieval and concatenation (joining together) when a message is to be formed; such synthesizers are only capable of speaking limited vocabulary in most cases. This method is known as copy synthesis, since the raw material from which the message is composed is derived from human speech [6], [7]. Alternatively, an electronic circuit is made to mimic the way human beings form speech with their vocal tracts, by constructing speech waves artifically, and since any speech sound unit can be constructed in this way, by rules, such synthesizers can easily cope with unlimited vocabulary. This method is known as synthesis-by-rule [3].

Although the speech generated by this method is poorer than that generated by copy synthesis, its ability to handle unlimited vocabulary with low memory requirement makes it very efficient for converting text to synthetic speech.

Many products capable of speaking unlimited vocabulary are now available as automated readers for the blind [8], [9] and communication aids for the nonvocal [9]. Some other promising applications of computer voice output are: a means of transmitting information from data bases to remote locations via the telephone line [9], [10]; a medium of communication with busy operators of systems who have to give all their visual attention to complicated processes and

Manuscript received May 16,1983; revised June 27, 1983.

The author was with the Department of Electronics, University of Southampton, England. He is now with the University of Ibadan, Ibadan, Nigeria.

would find displayed messages intolerable, as in the hazard warnings of an automobile [9]; and talking educational aids, like the Texas Instruments "Speak and Spell" [11].

The major condition imposed on the text-to-speech translator used in the system described in this paper is that it should be portable; thus large memory requirement cannot be tolerated. In addition, the system should be flexible and simple, independent to a large extent of any particular synthesizer, easy to interface with sources of text, capable of coping with unlimited vocabulary, and above all, it must be fast enough to operate at the normal speed of outputing speech intelligibly.

System Design

Human Factors

The blind definitely need talking machines as an alternative to Braille for reading textual information. Particularly, in schools and colleges for the blind, talking machines could be used for reading educational text. However, many of the commercially available reading aids for the blind are very expensive, and, in most cases, they are outside the means of the average person who needs them. For example, the Kurzweil Reading Machine which is now in many libraries costs about $30 000.

Therefore, in developing a reading aid for the blind, it is essential to target the aid to the financial capabilities of most of the potential users so that those who really need the aid can get it. Designing such aids around microprocessors and single-chip speech synthesizers can reduce the cost considerably. For example, the Votrax SC01 single-chip synthesizer costs less than $120, and the microcomputer-based circuitry plus the system software to drive the synthesizer, as well as any other input devices, may provide a reading aid for the blind for under $2000. This is what this project is aimed at achieving.

In addition, these aids should be easy and enjoyable to use. In order to target the facilities offered by the reading aid described in this paper, the design team consisted of the system designer, the author, working in consultation with some potential blind users who tested the system at each stage of development, and a human-factor expert who provided advice on the psychological aspects of the work, in keeping with the principles of man–machine systems. The final system evolved from a cycle of consultation, design, user trial and comments, and redesign. At the end of the project, a system evolved for reading texts in floppy-disk files which can be of educational use in schools and colleges for the blind.

System Hardware

The basic hardware of this system, illustrated in Fig. 1, consists of a microcomputer system (microcomputer, keyboard, and VDU) and a synthetic voice-output terminal. The source of text determines the type of any additional hardware. For example, if the text is pretyped into disks, then a disk drive will be necessary, and if the text is transmitted over the telephone line from a database on user request, then a decoder will be needed to decode the text into computer digestable form. An example of such system is the Videotex services [12]. If the text is coded in barcode [8], a barcode reader will be needed, and so on.

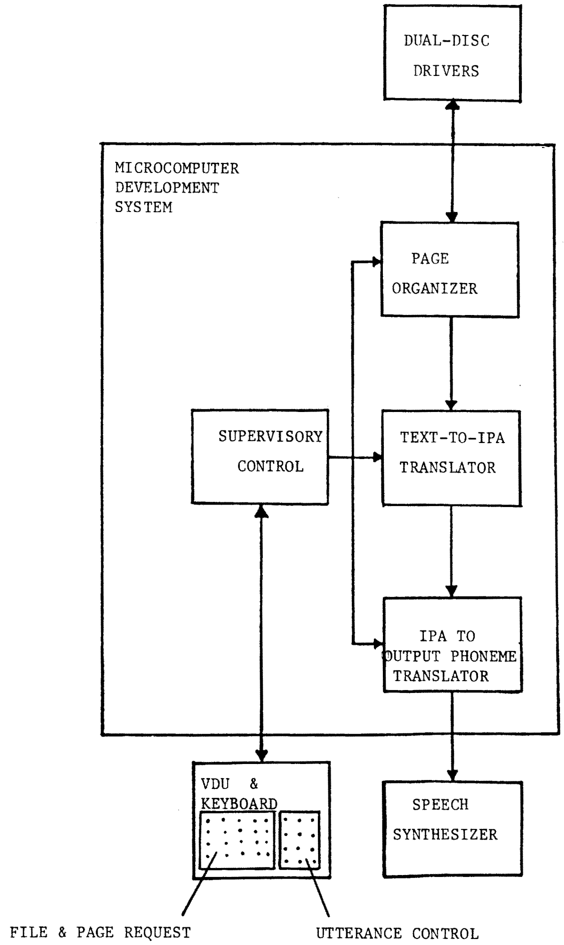

Fig. 1. Block diagram of the microcomputer-based reading aid for blind students.

In thinking about educational aids for the blind, many schools and colleges readily come to mind; and many of them might already have floppy disk operating microcomputer systems being used for other work. The system proposed in this paper will not require any costly additional hardware other than a simple commercial phonemic synthesizer. The text must have been pretyped into floppy disks from where the system reads any text file requested by a user.

Such a system has been developed with the Cromemco Z-2 microcomputer, dual floppy disk drives, a keyboard, a visual display unit (VDU), and a commercial speech synthesizer. To provide variety, two versions of the system were designed, each speaking through a particular synthesizer. The two speech synthesizers used are Microspeech 2 [13], and another one built purposely for the system, the Votrax SC-01 single-chip synthesizer [1].

Moreover, the design approach makes it easy to adapt the system to drive any other phonemic synthesizer with few alterations. This project started with Microspeech2 because it was readily available and was at that time relatively cheap compared to some synthesizers costing many thousands of dollars. But when the Votrax SC-01 chip became available, a similar system was designed to drive it.

Microspeech2, bought as a plug-in unit, plugs directly into an RS232 serial port of the microcomputer; there is no additional hardware needed to use this synthesizer. On the other hand, the Votrax SC01 single-chip synthesizer requires an interface circuitry to be driven by a microcomputer. In addition, it requires an audio amplifier circuit to output the speech signal via a loudspeaker. The detail of the circuit of the voice-output unit built around Votrax SCO1 is fully described by Ciarcia [1].

System Software

The system software was written in assembly language of a microprocessor because speed of translation must be very fast in this case. It occupies about 36 kbytes of memory which is well within the capacity of many microcomputers.

The software for converting texts in disk files into synthetic speech is organized in modular fashion, the three modules being

1) supervisory control
2) page retrieval and reorganization
3) text translation into synthesizer phonemes.

Within this scheme all the synthesizer utterances are made under the supervisory control after they have been formed into appropriate phonemes. The utterances could be of the text itself, or be internally generated instructions for the guidance of the user who has, it must be remembered, no other means of gaining any feedback from the system. For the purpose of discussion, however, the supervisory module shall be discussed last.

Page Retrieval and Reorganization

A file of text is called by typing the file name after which the "RETURN" key is pressed. This module finds the file and examines it in case it is larger than the text buffer, in which case it cannot be processed at once, and arrangement is made to read the file page by page. But if the size of the file is within the capacity of the text buffer, it is simply copied into the buffer and processed. In processing the page, all lines are terminated with "end-of-line" markers, and the whole text is terminated with an "end-of-text" marker.

All words in the page made up of alphanumeric characters are broken into two homogenous groups consisting of either alphabetic or numeric characters. For example, the compound word "page192" is broken into "page 192." All unnecessary spaces separating paragraphs or lines are removed, as well as all nonalphanumeric characters normally used to present the page in visually attractive and informative ways to the sighted reader. Some examples are the characters used to underline page title or important words and graphics. After the page has been processed as outlined above, control is passed to the text translator.

Text-to-Speech Translator

After processing under supervisory control, segments of the text are presented to the text translator, which first translates it to the International Phonetic Alphabets (IPA) with the aid of some letter-to-phonetic sound rules. These translation rules are based on Elovitz *et al.* [14] and cover all alphanumeric characters together with mathematical and punctuation signs. The rules are arranged in a set of tables in such a way that a character and its surrounding substrings may be rapidly matched to the appropriate phonemic (IPA) string which is then inserted into an intermediate phoneme buffer.

The IPA phonemes are then translated into the synthesizer phonemes, and inserted into the output phenome buffer. The translation software operates at a fast speed, and translates a sentence of about 120 characters in 1 s. In this two-stage translation technique, only the IPA-to-synthesizer translation software would need replacement whenever the synthesizer is changed; this is quite easy.

In order to provide satisfactory translation of English words within the capacity of a microprocessor-based system, the rule tables have been developed to apply to the most commonly used words taken from a list in which they are ranked in the order of their frequency of occurrence. The system has been tested against the first 7500 words in this list.

Also during the translation, necessary pauses are provided at punctuations, and in particular when "." is encountered the system checks whether it is a decimal point when it is pronounced as "point"; it is skipped when it occurs in initials like "Mr. R. Y. John" or used as the end-of-sentence marker. When the text segment is translated, the text translator informs the supervisory control which tests whether the output phoneme buffer is full, and transmits its contents to the synthesizer to speak. Otherwise, the system control presents and the next text segment to the translator to work on.

Supervisory Control

The control module provides the user with information about the state of the system, as "page not found," "selected file is empty," "end of page," etc., as well as providing control over the synthesizer output. On system startup, the user is invited to select a file, which is then searched for in the disk. If it is not found, the user is informed, and when found the system informs the user with a phrase like "file selected is—FILENAME." Similarly, when the end of a page is reached the user is informed, and he/she has the choice of reading the page again or selecting another one. Also, when accessing a long file, the user can terminate it at any stage to select another file.

The next big problem is how the blind person reads through a page of text independently! One fact is clear about synthetic speech—the highest speech quality realized so far (especially from phonemic synthesizers) is not as perfect as that of human speech. Consequently, a listener is not likely to understand anything the machine says for the first time, and therefore the facility for repeating any utterance is provided.

In addition, it is well known that there is a limit to the amount of information that can be held in human short-term memory [15]. Therefore, any sentence or line read by the system can be heard again, phrase by phrase, or word by word, and any word not clear can be spelled . Moreover, the user can switch between sentence and line modes as many times as he/she likes within a page.

In designing this system, the observation of potential blind users was incorporated into the exercise right from the beginning, and their comments after using the system at each stage contributed to the implementation of most of the speech-output controls. At a particular stage, it was observed that the blind person sometimes mistakenly selects next sentence (or line) and would therefore like to go back to the previous sentence (or line); so this provision was added to the controls.

All the controls are implemented in such a way that when a mode is selected with a key, the same key is used to select next utterance in the same mode. For example, when a sentence is to be read word by word, and the user presses the key for the word mode for the first time, the first word is automatically read out. When the key is pressed again, next word is spoken, and so on till the end of the sentence. In addition, it is possible to get out of a mode at any point desired and enter another mode. As such, one does not, for example, have to stay in the word mode until the end of the sentence if this is not desired.

The user can exit from a page at any time, obtain a fresh page, or leave the system completely. The user therefore has to gain familiarity with 10 operations, which are currently implemented by the numerical keys 0–9 on the VDU keyboard. Leaving the system, the eleventh operation, is effected with Control-C. Any other key such as ESCAPE can be used, but a two-key operation such as Control-C protects the system from being accidentally switched off. All of the 11 operations are listed as follows:

Sentence:	sentence mode
Line:	line mode
Back:	read previous sentence (or line)
Phrase:	phrase mode
Word:	word mode
Spell:	spell current word
Repeat:	repeat current utterance
Restart:	restart current mode of speaking
End:	leave current page and get next
Terminate:	terminate access to current text file and return file select control to user
Leave:	leave system.

All of these speech control processes are performed by moving pointers in the reformatted page store, rather than the phoneme buffer, since the translation process is sufficiently fast to incur no additional time penalty. A sentence of 120 characters is translated into speech in 1 s. Thus, when a key is pressed for a sentence, line, phrase, word, or the spelling of a word, the speech output is almost instantaneous. This makes the speech sound realistic.

At the start of translating a sentence or line, a pointer P(current) is set at the beginning of the statement. Also at the end of translation, another pointer P(next) is set at the beginning of the next statement.

So when the user requests that the sentence (or line) be read in word mode, the supervisory control calls on a subroutine WORD which sets a second pointer P(word) to the start of the first word. It then translates this word into speech to be uttered by the synthesizer. Next, WORD moves the pointer, P(word) to the beginning of the next word which is also spoken. In this way, the whole sentence or line is read out word by word. If at any point the user requests for the spelling of a word just spoken, then WORD calls on yet another subroutine SPELL which starts spelling from the current position of P(word) up to an end-of-word marker.

To read a sentence or line in phrase mode, a subroutine PHRASE is called to operate in a pattern similar to WORD by moving a pointer P(phrase) within the text buffer. Comma, colon, semicolon, etc., are used as end-of-phrase markers. Spelling is only active within word mode because it is unnecessary to spell a phrase, line, or sentence. When reading a sentence or line in word or phrase mode, the pointer P(next) is used to indicate the end of the last word or last phrase, as the case may be.

When the user presses the BACK key to request the previous sentence or line, the pointer P(current) is moved back to the beginning of this statement, from where the translation starts again. When the REPEAT key is pressed, the content of the output phoneme buffer is sent to the synthesizer to be spoken again. However, the response of the system to the RESTART key depends on the mode of reading as follows:

Mode	Response to Pressing RESTART Key
Sentence or Line	P(current) is moved to the beginning of current page, and the first sentence (or line) spoken.
Phrase	P(phrase) is moved to the beginning of the current sentence (or line) and first phrase spoken.
Word	P(word) is moved to the beginning of the current sentence (or line) and first word spoken.
Spell	(word) moved to the beginning of the current word and spelled again.

When the END key is pressed, the system exits from the current page to read the next page into the text buffer. If the current page is the last page of the file being read, then the file selection control is passed on to the user. The TERMINATE key initiates the closure of the current file and the return of the file selection control to the user. When the LEAVE key is pressed, the system closes current file and switches off all operations.

Involving potential users in evaluating the system at various stages of development has helped immensely in steering the system controls to the needs of the users. This is in keeping with the human factor principles of man–machine interactions which stress the need to design a system which is easy for users to use [16].

System Evaluation

It is probably clear that the facilities of the system described in this paper have not been developed in total isolation from the potential users. But despite this, the final system was fully evaluated by a wider circle of users in order to determine which was the better of the two speech synthesizers used, and also to observe in reality the ease with which the system can be used.

The performance of the system was evaluated by two listening tests involving both blind and sighted users. In the first test, 16 subjects used the system to read special words and sentences, as well as stories on which they answered some questions. After the test, each subject was asked to assess the system. The result showed that the system performed far better when speaking via Votrax SCO1 than via Microspeech2. Coupled with the fact that Microspeech2 is more expensive and about twice the size of the Votrax SCO1 based unit, Votrax SCO1 is therefore the right choice for the proposed system.

In the second test, when the system speaking via Votrax SCO1 only was used to read everyday sentences by a new set of listeners, all of them got used to the system in about 30 min, and after about an hour they all understood more than 90 percent of the words of the sentences. Eight blind and eight sighted native British English speakers, and six nonnative English speakers took part in this second test. Their performances are illustrated in Figs. 2 and 3.

In order to monitor the manner in which the subjects used the system to read pages of text throughout the evaluation experiments, the system was modified to operate as follows.

When any page of text requested by a subject was received, the page title or the first line was copied into a data record, as well as the time received. Also, when a speech-output key was pressed, the key and the time when pressed, as well as the text read as a result of pressing the key were recorded. All data were automatically recorded into a floppy disk file in the following format:

DATA = [Key]:[Time]:[Text read with key]

Time = (hour):(minute):(second):(second/10)

From this data the time spent on various parts of the page was estimated as an indication of the difficulties encountered in using the system. On the whole, all subjects were able to use the system with little difficulty.

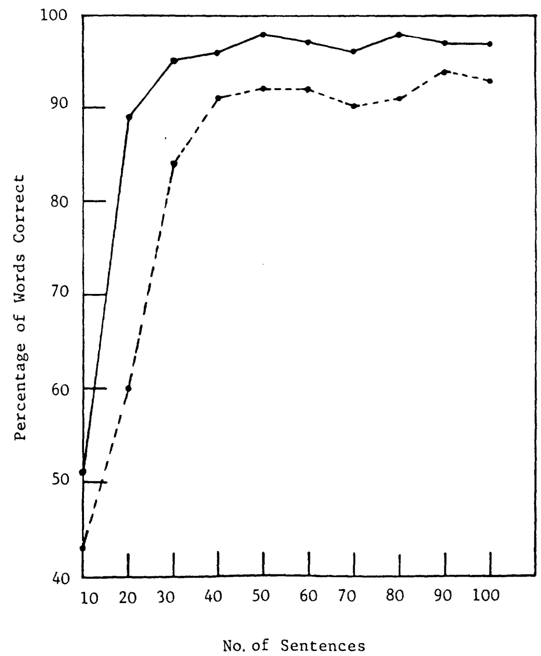

Fig. 2. Percentage of words correctly perceived for successive ten-sentence sets. ——• Eight native British English speakers. ---• Six nonnative English speakers.

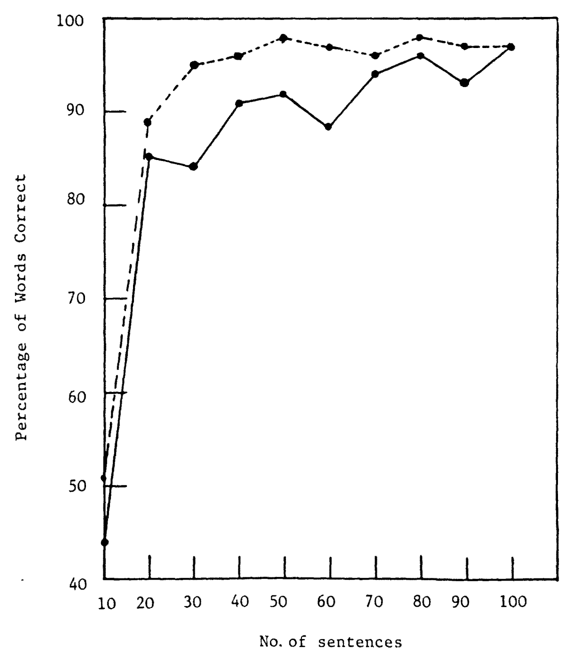

Fig. 3. Percentage of words correctly perceived for successive ten-sentence sets. ——•Eight blind native British English speakers. ---•Eight sighted native British English speakers.

Conclusion

The technique used in designing a microcomputer-based system for automatic conversion of text into synthetic speech has been described in this paper. In particular, this system is designed to be used as a reading aid by blind students.

It is a well known fact that many blind people after appropriate training can conveniently operate the keyboard of conventional microcomputers. Therefore, selecting a file by typing the file name, as well as controlling the speech output of the system with keys on the keyboard should not present a major problem.

However, handicaps could be of a multiple nature and the blind user may not be able to use his/her hands to operate a normal keyboard properly, as for example, in the case of stroke, arthritis, or any other ailments in which paralysis of upper limbs may occur.

In such cases, special input devices [17] must be considered for selecting text and for controlling the speech output of the system. The type of disability would indicate the type of input device. When modified conventional keys [17] can, however, be conveniently used by a handicapped person, this alternative should be considered first. This is because special input devices, unlike conventional keys, are very expensive and require special software to decode their input data.

The system described in this paper is a practicable reading aid for blind students. Many blind users of this system have been able to use it confidently after about 3 h of training. However, Braille is still the right choice in reading text which contains many diagrams or maps. But apart from this, the conversion of the text into speech is adequate, with the added bonus of faster reading speed.

ACKNOWLEDGMENT

The author wishes to thank all members of the Man–Machine Research Group of the Department of Electronics, University of Southampton, Southampton, England, for their advice and general discussions. Thanks are also due to B. Payne and J. Deaper of the Hampshire Association for the Care of the Blind, and to numerous volunteer subjects for their assistance in the evaluation of the system described in this paper.

REFERENCES

[1] S. Ciarcia, "Build an unlimited-vocabulary speech synthesizer," *BYTE*, p. 38, Sept. 1981.
[2] W. Smith and S. B. Crook, "Phonemes, allophones, and LPC team to synthesize speech," *Electron. Des.* p. 121, July 1981.
[3] J. L. Flanagan, C. H. Coker, L. R. Rabiner, R. W. Schafer, and N. Umeda, "Synthetic voices for computers," *IEEE Spectrum*, p. 22, Oct. 1970.
[4] O. R. Omotayo, "Microprocessors: Problems of getting started," *IEEE Trans. Educ.*, p. 179, May 1981.
[5] R. N. Noyce and M. E. Hoff, "A history of microprocessor development at Intel," *IEEE Micro*, p. 8, Feb. 1981.
[6] R. Gunarwardana, "General purpose speech synthesizer peripheral using LPC," presented at the IEE Colloq. Comput. Gen. Speech, London, England, Jan. 1981.
[7] R. Allan, "Speech input and output teem with developments," *Electron. Des.*, p. 67, May 1982.
[8] J. T. Powers, "Text publishing and reading system for the handicapped," *IBM Tech. Disc. Bull.*, p. 2421, Nov. 1977.
[9] E. J. Lerner, "Products that talk," *IEEE Spectrum*, p. 32, July 1982.
[10] J. D. Andrews, "Automatic voice systems for telecommunication," presented at the IEEE Colloq. Comput. Gen. Speech, London, England, Jan. 1981.
[11] R. Wiggins and L. Brantingham, "Three-chip system synthesizes human speech," *Electronics*, p. 109, Aug. 1978.
[12] O. R. Omotayo, "Synthetic speech output for electronic information services," Ph.D. dissertation, Univ. Southampton, Southampton, England, 1983.
[13] R. Monkhouse and T. Orr, *Microspeech2 Operators Handbook*, Costronics Electronics, Hillingdon, United Kingdom.
[14] H. S. Elovitz, R. Johnson, A. McHugh, and J. E. Shore, "Letter-to-sound rules for automatic translation of English text to phonetics," *IEEE Trans. Acoust., Speech, Signal Processing*, p. 446, Dec. 1976.
[15] M. J. Underwood, "Systems aspects of using speech output," presented at the IEE Colloq. Comput. Gen. Speech, London, England, Jan. 1981.
[16] H. Simpson, "A human-factors style guide for program design," *BYTE*, pp. 108–132, Apr. 1982.
[17] N. L. Staisey, J. W. Tombauch, and R. F. Dillon, "Videotex and the disabled," *Int. J. Man-Machine Studies*, vol. 17, p. 33, 1982.

Oluwole Remi Omotayo (M'80) received the B.Sc. degree in physics from Ahmadu Bello University, Zaria, Nigeria, and the M.Sc. degree in electronics from the University of Southampton, Southampton, England.

He was a lecturer at the Polytechnic, Ibadan, Nigeria, from 1974 to 1979, and was Ag. Head of the Department of Electrical/Electronic Engineering of Ogun State Polytechnic, Abeokuta, Nigeria, from January to September 1980. He was on study leave to the University of Southampton, Southampton, England, from October 1980 to September 1983, where he completed his Ph.D. research. He joined the Department of Electrical Engineering of the University of Ibadan in Nigeria in September 1983.